OPENACS ON DOCKER

Managing multiple OpenACS Sites with Docker Compose
REQUIREMENTS/GOALS

✴ Have a fully off-line development environment
  ➡ Code at the cabin

✴ Minimize RAM and CPU usage for work on a laptop
  ➡ Single (usually) database server

✴ Development instances for Multiple OpenACS sites
  ➡ Easily run only needed instances
  ➡ Easily synchronize code with production and/or other developers
SOLUTION

- Docker Compose projects/environments
- Postgres (or Oracle) and NaviServer in separate containers
  - Use existing Postgres or PostGIS containers from Docker Hub
  - Build NaviServer container from Dockerfile or pull from Docker Hub
- Tools to manage code checkout and synchronization of many
  - Developed git-sync.tcl and friends
- Tools to manage database creation/load/modification
  - Created refresh-site-db.sh to manage the database for a site
    Can update parameter values in a cloned/restored database, before starting NS
version: '3.2'
networks:
  default:
    name: lsw-network
    external: true
services:
  postgres15:
    image: postgis/postgis:15-3.3
    container_name: lsw-pgdb15
    restart: unless-stopped
    volumes:
      - ~/dev/docker/lsw/pg15data:/var/lib/postgresql/data
      - ~/dev/docker/lsw/pg15conf:/var/lib/postgresql/conf
      - ~/dev/docker/lsw/pg15log:/var/lib/postgresql/log
    ports:
      - 127.0.0.1:5433:5432
    environment:
      - POSTGRES_USER=nsadmin
      - PGUSER=nsadmin
      - POSTGRES_PASSWORD=df8auf48u48of3ChangeMe
      - POSTGRES_DB=nsadmin
<table>
<thead>
<tr>
<th>lswbase:</th>
<th>volumes:</th>
</tr>
</thead>
<tbody>
<tr>
<td># build:</td>
<td>- ~/dev/docker/lsw/lswbase/etc:/web/etc</td>
</tr>
<tr>
<td>#   context: ./builds/naviserver</td>
<td>- ~/dev/docker/lsw/lswbase/logs:/web/logs</td>
</tr>
<tr>
<td>#   dockerfile: Dockerfile</td>
<td>- ~/dev/docker/lsw/lswbase/files:/web/files</td>
</tr>
<tr>
<td>#   target: nsbfinalo</td>
<td>- ~/dev/docker/lsw/lswbase/openacs:/web/site</td>
</tr>
<tr>
<td>image: logansw/naviserver:v1.12</td>
<td>- ~/dev/docker/lsw/lswbase/etc/config.tcl:/usr/local/ns/conf/config.tcl</td>
</tr>
<tr>
<td>container_name: lsw-lswbase</td>
<td>- ~/dev/docker/lsw/certs:/web/certs</td>
</tr>
<tr>
<td>restart: unless-stopped</td>
<td>ports:</td>
</tr>
<tr>
<td>stdin_open: true</td>
<td>- 80:80</td>
</tr>
<tr>
<td>tty: true</td>
<td>- 443:443</td>
</tr>
<tr>
<td>entrypoint: /web/etc/nsd-container-start.sh</td>
<td>depends_on:</td>
</tr>
<tr>
<td>command: &quot;/usr/local/ns/bin/nsd -u nsadmin\</td>
<td>- postgres15</td>
</tr>
<tr>
<td>-g web -i -t /web/etc/config.tcl\</td>
<td></td>
</tr>
<tr>
<td>-b 0.0.0.0:80,0.0.0.0:443&quot;</td>
<td></td>
</tr>
</tbody>
</table>
#!/bin/bash
umask 002

counter=0
until psql -U nsadmin -h lsw-pgdb15 lswbase -c "select 1" &> /dev/null
do
  let "counter++"
  sleep 3
  if [ $counter -ge 20 ]; then
    echo "Failed to connect to database 'lswbase' on lsw-pgdb15"
    exit 1
  fi
done

echo "postgres database 'lswbase' is available. Starting naviserver."
exec "@"
NOTABLE

✴ Set COMPOSE_PROJECT_NAME in a .env file in the directory with docker-compose.yml
✴ cd to the directory with the .env file before running “docker compose” commands
✴ The depends_on directive only waits until the Postgres container is started. Postgres may not be accepting connections when the NS container starts. Hence the nsd-container-start.sh script
✴ The build directive, when uncommented, takes precedence over the image directive
✴ container_names can be used as a hostname which resolves to the Docker-assigned IP address for the container
TOOLS

- **Git info/compare/sync**
  - **git-info.tcl**
    Creates a git-info file containing git info for all repositories below the Current Working Directory (CWD)
  - **git-diff.tcl**
    Compares git repositories below the CWD to those in a git-info file
  - **git-sync.tcl**
    Clone or update git repositories under the CWD to match those in a git-info file
  - **git-status.tcl**
    Show results of “git status” for each repository under the CWD
WANT MORE?

✴ Full instructions to create an LSW stack are available on the “Docker Install” page at: https://logansw.atlassian.net/wiki/spaces/LSWS

✴ git-sync and friends are available on our bitbucket server at: https://bbs.logansw.com/projects/TOOLS/repos/git-sync

✴ Other Open Source tools and packages are available on the Bitbucket server
  ➡ Some are just copies of OpenACS core packages with tags added for git-sync

✴ ZFS—It’s where your data belongs
  (No presentation by me is complete without a reference to ZFS)
DISCLAIMERS

✴ There will be some differences from a stock NaviServer/OpenACS install
✴ PostgreSQL driver mod to support bind variable dereferencing of array vars
✴ Minor tweaks to OpenACS core - for backward compatibility with legacy packages
✴ OS utilities you may not care about in the NaviServer container
✴ LSW ecosystem packages support PostgreSQL only
  Mainly due to extensive use of tables with Range Types
✴ This is a work-in-progress. Feedback is welcome.